

|  |  |
| --- | --- |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  федеральное государственное бюджетное образовательное учреждение высшего образования  **«МОСКОВСКИЙ АВТОМОБИЛЬНО-ДОРОЖНЫЙ ГОСУДАРСТВЕННЫЙ**  **ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ (МАДИ)»** |  |

**Кафедра «Высшая математика»**

**Лабораторная работа № 2**

по дисциплине

«Программирование для ЭВМ»

на тему:

**«**Работа с файлами**»**

**Выполнил:**

Учебная группа: 1бПМ   
ФИО: Греча К. П.

Подпись: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Руководитель лабораторной работы:**

Должность: старший преподаватель

Звание: б/з

ФИО: Кутейников И. А.

Подпись: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_г.

Цель и задачи:

Реализовать простейший файловый менеджер. (Для Java использовать пакеты

java.io|java.nio, класс File|Files). Программа должна обладать следующим

функционалом:

1 Выбор файла или каталога для работы;

2 Вывод абсолютного пути для текущего файла или каталога;

3 Вывод содержимого каталога;

4 Вывод всей возможной информации для заданного файла;

5 Изменение имени файла или каталога;

6 Создание нового файла или каталога по заданному пути;

7 Создание копии файла по заданному пути;

8 Вывод списка файлов текущего каталога, имеющих расширение,

задаваемое пользователем;

9 Удаление файла или каталога;

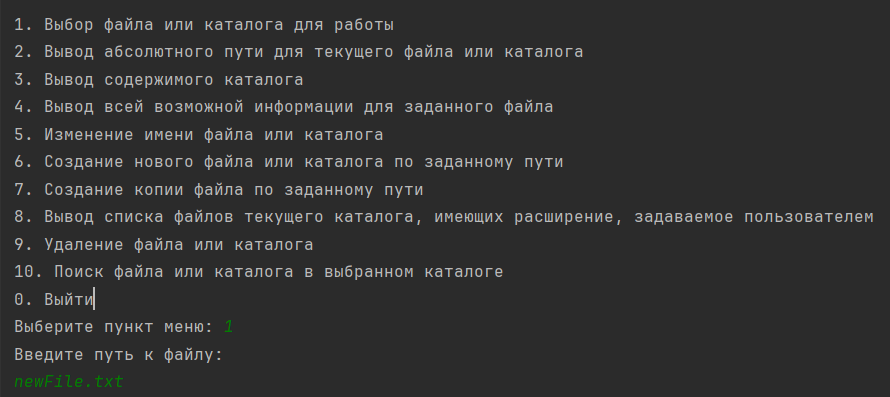
10.Поиск файла или каталога в выбранном каталоге;

Алгоритм и меню файлового менеджера:

import java.io.\*;  
import java.util.Scanner;  
import java.nio.file.StandardCopyOption;  
import java.nio.file.Path;  
import java.nio.file.Files;  
  
public class Main {  
 public static void main(String[] args) {  
 Scanner in = new Scanner(System.*in*);  
 int variant;  
 File f1=null;  
 while (true) {  
 System.*out*.println("\n1. Выбор файла или каталога для работы ");  
 System.*out*.println("2. Вывод абсолютного пути для текущего файла или каталога");  
 System.*out*.println("3. Вывод содержимого каталога");  
 System.*out*.println("4. Вывод всей возможной информации для заданного файла");  
 System.*out*.println("5. Изменение имени файла или каталога");  
 System.*out*.println("6. Создание нового файла или каталога по заданному пути");  
 System.*out*.println("7. Создание копии файла по заданному пути");  
 System.*out*.println("8. Вывод списка файлов текущего каталога, имеющих расширение, задаваемое пользователем");  
 System.*out*.println("9. Удаление файла или каталога");  
 System.*out*.println("10. Поиск файла или каталога в выбранном каталоге");  
 System.*out*.println("0. Выйти");  
  
 System.*out*.print("Выберите пункт меню: ");  
 try {  
 variant = Integer.*parseInt*(in.nextLine());  
 } catch (NumberFormatException e) {  
 System.*out*.println("Неверный выбор");  
 continue;  
 }  
 switch (variant) {  
 case 1:  
 System.*out*.println("Введите путь к файлу: ");  
 String f=in.nextLine();  
 f1=new File(f);  
 break;  
  
 case 2:  
 if (f1!=null) {  
 System.*out*.println("Абсолютный путь к файлу: " + f1.getAbsolutePath());  
 }  
 break;  
  
 case 3:  
 if (f1 != null && f1.isDirectory()) {  
 System.*out*.println("Содержимое каталога:");  
 File[] files = f1.listFiles();  
 for (File f10 : files) {  
 System.*out*.println(f10.getName());  
 }  
 } else {  
 System.*out*.println("Необходимо выбрать каталог для просмотра содержимого.");  
 }  
 break;  
  
 case 4:  
 if (f1 != null) {  
 if (f1.exists()) {  
 System.*out*.println("Информация о файле: ");  
 System.*out*.println("Имя файла: " + f1.getName());  
 System.*out*.println("Тип файла: " + (f1.isDirectory() ? "Директория" : "Файл"));  
 System.*out*.println("Absolute Path " + f1.getAbsolutePath());  
 System.*out*.println("Parent "+f1.getParent());  
 System.*out*.println(f1.canWrite()?"can write":"can't write");  
 System.*out*.println(f1.canRead()?"can read":"can't read");  
 System.*out*.println("File size " + f1.length() + " байт");  
 System.*out*.println("Last modified " + f1.lastModified());  
 } else {  
 System.*out*.println("Файл не существует");  
 }  
 } else {  
 System.*out*.println("Файл не выбран ");  
 }  
 break;  
  
 case 5:  
 if (f1 != null) {  
 System.*out*.println("Введите новое имя файла или каталога:");  
 String newName = in.nextLine();  
 String newPath = f1.getParent() + File.*separator* + newName;  
 File newFile = new File(newPath);  
 if (f1.renameTo(newFile)) {  
 System.*out*.println("Имя файла или каталога успешно изменено ");  
 f1 = newFile;  
 } else {  
 System.*out*.println("Не удалось изменить имя файла или каталога ");  
 }  
 } else {  
 System.*out*.println("Файл не выбран ");  
 }  
 break;  
  
 case 6:  
 System.*out*.println("Введите путь и название нового файла или каталога:");  
 String newFilePath = in.nextLine();  
 File newFile = new File(newFilePath);  
 try {  
 if (newFile.createNewFile()) {  
 System.*out*.println("Файл или каталог успешно создан ");  
 } else {  
 System.*out*.println("Не удалось создать файл или каталог ");  
 }  
 } catch (IOException e) {  
 System.*out*.println("Ошибка при создании файла или каталога: " + e.getMessage());  
 }  
 break;  
  
 case 7:  
 if (f1 != null) {  
 System.*out*.println("Введите путь для копии файла: ");  
 String copyPath = in.nextLine();  
 Path sourcePath = f1.toPath();  
 Path Path1 = new File(copyPath + File.*separator* + f1.getName()).toPath();  
 try {  
 Files.*copy*(sourcePath, Path1, StandardCopyOption.*REPLACE\_EXISTING*);  
 System.*out*.println("Копия файла успешно создана ");  
 } catch (IOException e) {  
 System.*out*.println("Ошибка при создании копии файла: " + e.getMessage());  
 }  
 } else {  
 System.*out*.println("Файл не выбран ");  
 }  
 break;  
  
 case 8:  
 System.*out*.println("Введите каталог из которого хотите вывести файлы: ");  
 String c=in.nextLine();  
 File c1=new File(c);  
 System.*out*.println("Введите расширение файлов которые хотите вывести: ");  
 String r = in.nextLine();  
 FilenameFilter filter = new FilenameFilter() {  
 @Override  
 public boolean accept(File dir, String name) {  
 return name.endsWith(r);  
 }  
 };  
 File[] arrayf = c1.listFiles(filter); // Применяем фильтр здесь  
  
 if (arrayf != null && arrayf.length > 0) {  
 System.*out*.println("Файлы с расширением " + r + ":");  
 for (File file : arrayf) {  
 System.*out*.println(file.getName());  
 }  
 } else {  
 System.*out*.println("Файлы с расширением " + r + " в данном каталоге не найдены.");  
 }  
 break;  
 case 9:  
 if (f1 != null) {  
 if (f1.isFile()) {  
 if (f1.delete()) {  
 System.*out*.println("Файл успешно удален");  
 } else {  
 System.*out*.println("Не удалось удалить файл");  
 }  
 } else if (f1.isDirectory()) {  
 File[] files = f1.listFiles();  
 if (files != null) {  
 for (File file : files) {  
 if (file.isDirectory()) {  
 *deleteDirectory*(file);  
 } else {  
 if (file.delete()) {  
 System.*out*.println("Файл " + file.getName() + " успешно удален");  
 } else {  
 System.*out*.println("Не удалось удалить файл " + file.getName());  
 }  
 }  
 }  
 }  
 if (f1.delete()) {  
 System.*out*.println("Каталог успешно удален");  
 } else {  
 System.*out*.println("Не удалось удалить каталог");  
 }  
 } else {  
 System.*out*.println("Объект не является ни файлом, ни каталогом");  
 }  
 } else {  
 System.*out*.println("Введите файл или каталог");  
 }  
 break;  
  
 case 10:  
 System.*out*.println("Введите название файла или каталога который хотите найти: ");  
 String search=in.nextLine();  
 File fsearch=new File(search);  
 if(fsearch.exists()){  
 System.*out*.println("Файл найден"+"\n"+"Его путь: "+fsearch.getAbsolutePath());  
 }else  
 System.*out*.println("Файл не найден");  
 break;  
  
 default:  
 System.*out*.println("Вы ввели неверный пункт меню\nПопробуйте снова ");  
 }  
 }  
 }  
 private static void deleteDirectory(File directory) {  
 File[] files = directory.listFiles();  
 if (files != null) {  
 for (File file : files) {  
 if (file.isDirectory()) {  
 *deleteDirectory*(file);  
 } else {  
 file.delete();  
 }  
 }  
 }  
 directory.delete();  
 }  
}

**Результат:**

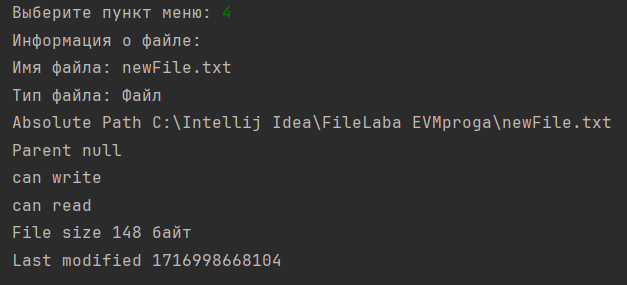
Выбор файла для работы:



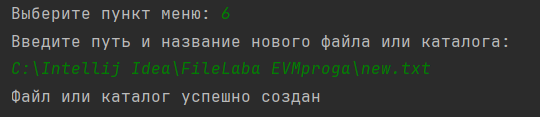
Вывод абсолютного пути:



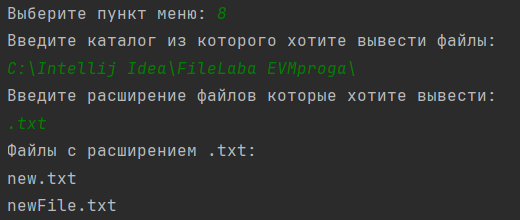
Вывод всевозможной информации о файле:



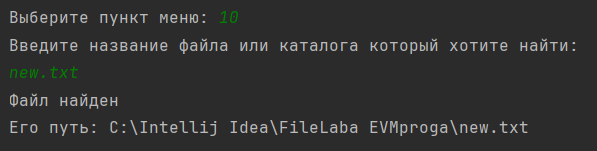
Создание нового файла:



Вывод списка файлов каталога, имеющих введенное расширение:



Поиск файла в выбранном каталоге:



**Заключение:**

В практической работе реализуется файловый менеджер. Были изучены навыки работы с файлами, а именно нахождение файла с заданным расширением, удаление файла из каталога, вывод всевозможной информации о текущем файле.